



**Queensland University of Technology**  
Brisbane Australia

This is the author's version of a work that was submitted/accepted for publication in the following source:

[Alwadain, Ayed, Fielt, Erwin](#), Korthaus, Axel, & [Rosemann, Michael](#) (2011) Where do we find services in enterprise architectures? A comparative approach. In *Proceedings of the 22nd Australasian Conference on Information Systems (ACIS 2011)*, Sydney Law School, Sydney, NSW.

This file was downloaded from: <http://eprints.qut.edu.au/47469/>

© Copyright 2011 [please consult the author]

**Notice:** *Changes introduced as a result of publishing processes such as copy-editing and formatting may not be reflected in this document. For a definitive version of this work, please refer to the published source:*

## Where Do We Find Services in Enterprise Architectures? A Comparative Approach

Ayed Alwadain  
School of Information Systems  
Queensland University of Technology  
Brisbane, Australia  
Email: a.alwadain@student.qut.edu.au

Erwin Fielt  
School of Information Systems  
Queensland University of Technology  
Brisbane, Australia  
Email: e.fielt@qut.edu.au

Axel Korthaus  
School of Management and Information Systems  
Victoria University  
Melbourne, Australia  
Email: axel.korthaus@vu.edu.au

Michael Rosemann  
School of Information Systems  
Queensland University of Technology  
Brisbane, Australia  
Email: m.rosemann@qut.edu.au

### Abstract

*In recent years, enterprise architecture (EA) has captured growing attention as a means to systematically consolidate and interrelate diverse IT artefacts in order to provide holistic decision support. Since the emergence of Service-Oriented Architecture (SOA), many attempts have been made to incorporate SOA artefacts in existing EA frameworks. Yet the approaches taken to achieve this goal differ substantially for the most commonly used EA frameworks to date. This paper investigates and compares five widely used EA frameworks in the way they embrace the SOA paradigm. It identifies what SOA artefacts are considered to be in the respective EA frameworks and their relative position in the overall structure. The results show that services and related artefacts are far from being well-integrated constructs in current EA frameworks. The comparison presented in this paper will support practitioners in identifying an EA framework that provides SOA support in a way that matches their requirements and will hopefully inspire the academic EA and SOA communities to work on a closer integration of these architectures.*

### Keywords

Enterprise Architecture, EA, Service-Oriented Architecture, SOA, Service-orientation, Integration.

### INTRODUCTION

During the last decades, modern enterprises have invested heavily in a multitude of IT artefacts. A commonly accepted approach to dealing with the resulting complexity and to improving overall business/IT alignment revolves around the specification of Enterprise Architecture (EA). EA is a holistic perspective incorporating all artefacts of an enterprise such as business, organisational, application, information, data and infrastructure artefacts (Buckl et al. 2010). It describes and models elements of organisations and shows how they are organised and connected, and how they function as a whole. EA is not a physical artefact in itself; rather, it produces the artefacts (e.g., models) that illustrate the enterprise's present and desired future structure (Seppänen 2008). In general, EA should be organised in a manner that explains the structure, behaviour and properties of the enterprise. It specifies the elements that constitute the enterprise. Further, EA elements and artefacts need to be applicable for a broad range of enterprises and government agencies (Winter and Fischer 2007). However, there is a lack of consensus on EA terminologies, concepts, approaches and outcomes which leads to confusion in communications within the EA discipline and with EA stakeholders (Luo 2006; Mykhashchuk et al. 2011). Further, EA provides many benefits to enterprises such as aligning business and IT, improving organisational communications and information sharing, and reducing IT complexity (Tamm et al. 2011). In order to continue to deliver these benefits, EA frameworks themselves need to embrace change in ways that adequately consider the emerging new paradigms and requirements affecting EA, such as the paradigm of service orientation (Alwadain et al. 2010).

Service-Oriented Architecture (SOA) can be viewed from two perspectives: a narrow technical and a broader business/managerial view. Although most of the definitions are predominantly technical, recent publications have taken a broader perspective of SOA from a business viewpoint (Lee et al. 2010). From a technical perspective, SOA is a software architecture approach where the basic components of design and development are services (Kumar et al. 2007). However, SOA from the broader business perspective is defined by Engels et al. (2008) as a paradigm for structuring the business of an enterprise in the form of services which accordingly

drives the IT architecture. It provides a new way for enterprise engineering that covers both the organisational and technical aspects of an enterprise. This concept is based on the understanding of business capabilities as services (e.g. payment, fraud detection) down to the technical implementation of encapsulated software capabilities in terms of Web Services (Lee et al. 2010; Stein et al. 2008). Therefore, SOA principles and elements should be used in the conceptual modelling of EA (Gustas 2007).

Along the movement towards service-orientation, EA is important for managing the services landscape. Enterprise architects have to adapt their methodologies and concepts in order to manage the complex service architecture (Postina et al. 2010). Services are increasingly considered as one of the enterprise's essential assets which need to be considered and integrated in EA (Correia and Silva 2007; Khoshnevis et al. 2009). Correia and Silva (2007) emphasise the need to capture services within EA frameworks. They highlight the importance of an integrated and cohesive vision of services in EA in order to increase organisational agility. Further, Sanders et al. (2008) highlight the need to combine SOA with EA frameworks in order to describe a complete system. Grigoriu (2007) argues that SOA and EA are about the architecture of the enterprise. Thus, SOA and EA should be seen as complementary to each other rather than alternatives (Knippel and Skytte 2007; Seppänen 2008). SOA requires an intensive enterprise re-engineering effort which has consequences for the different EA layers such as process, applications and infrastructure. He argues that SOA implementation does not achieve its objectives outside the context of EA development, because SOA relies on EA "as-is" models and artefacts. EA is needed to plan the journey from an "as-is" to a "to-be" service-oriented EA before the implementation or adoption of SOA. When adopting SOA, the EA team should embrace SOA to increase the visibility and the impact of SOA (Paras et al. 2007). EA can be seen as an approach which describes the baseline and target architectures and assists in developing a roadmap for SOA adoption. Therefore, enterprises need an EA with supporting methods and tools in order to implement an architecture based on SOA and to take advantage of the new capabilities that SOA offers (Perko 2008). In addition, service-orientation descriptions can be employed for the semantic integration of both the dynamic and static aspects of EA frameworks (Gustas 2007).

Currently, selected EA frameworks provide support for service orientation and integrate SOA elements, e.g. TOGAF and DoDAF. However, there is inconsistency in what elements of SOA need to be captured and positioned in the different layers of EA. This is even aggravated by the observation that SOA standards are inconsistent in terms of definitions and expression and may not provide sufficient models required by practitioners to establish best practices for architecture (Sprott 2011). Erl (2005) argues that service orientation and SOA appear to be confusing terms. Viering et al. (2009) found that in regard to SOA, a key challenge is the lack of consistency with respect to emphasis, terminology and levels of abstraction employed to describe SOA design principles. SOA stakeholders have various concerns which need sufficient viewpoints to address these needs. Nevertheless, enterprise architecture frameworks are not yet adequately reflecting the shift towards SOA nor fully addressing the viewpoints of SOA stakeholders (Postina et al. 2010). A key challenge arising from the different approaches to integrate SOA and EA frameworks lies in addressing completely and correctly the SOA-related concerns of various stakeholders. An enterprise seeking to adopt SOA and document current and future SOA elements in EA will potentially run into difficulties when identifying and classifying a relevant and complete set of SOA elements for different stakeholders using EA framework that has not sufficiently addressed SOA concerns and characteristics. Thus, in order to address this challenge, the architectural descriptions referring to SOA elements in the current or future states of the EA need to represent and match the corresponding concerns of the stakeholders.

This paper addresses the aforementioned challenge by comparing widely used EA frameworks based on the identification of SOA elements in these frameworks and their positions in each framework. In doing so, this paper addresses the following research question: How do major Enterprise Architectures integrate these SOA elements?

The contribution of this paper is that it presents the first thorough analysis and comparison of EA frameworks in terms of their integration of SOA elements in order to better understand and integrate SOA in EA frameworks and to ensure that systems are built based on a comprehensive understanding of the concerns of the different stakeholders and assure the use of common terminology and content of SOA architectural descriptions (Greefhorst et al. 2006). Thus, in order to establish a common ground for the framework comparison, different EA frameworks were systematically studied. SOA elements were identified and each framework was assessed in terms of its capability to capture these elements, after which the frameworks were compared.

The remainder of the paper is structured as follows. Section two briefly reviews the literature on EA and SOA. Section three presents the research design of this study. Section four describes and analyses five widely used EA frameworks that support SOA. Section five compares and discusses their SOA integration. The last section presents conclusions and an outlook on future research.

## LITERATURE REVIEW

In recent years, Enterprise Architecture has gained a considerable importance in academia and practice. It is considered an important means to support enterprise transformation and to enable the managed evolution of organizations (Mykhashchuk et al. 2011). Lankhorst (2005) characterises EA as “a coherent whole of principles, methods, and models that are used in the design and realization of an enterprise’s organisational structure, business processes, information systems, and infrastructure”. In general terms, the enterprise framework approach defines how information technology is associated with enterprise business processes and outcomes, and describes relationships between technical and organisational elements of the enterprise (Weerakkody et al. 2007). EA can be considered as a means to address the complexity of modern enterprises (Seppänen 2008; Zachman 1987). It describes and models the organisations’ elements and shows how they are organised, connected and function as a whole.

Although a plethora of EA frameworks has been developed (Berrisford and Lankhorst 2009; Buckl et al. 2010; Sowa and Zachman 1992; The UK Department of Defence 2010b; US Department of Defence 2009), they all have a set of shared benefits. EAs improve the understanding of an enterprise’s business and information systems landscape and support holistic decision making (Franke et al. 2009). EA documents and helps maintain the enterprise’s current IT and business landscapes and its related plans, and facilitates the development of the future state. At a high level of abstraction, EA is a communication method between stakeholders (Chen et al. 2008). It facilitates the communication of essential constituents of an enterprise to different stakeholders by allowing different viewpoints and alternative levels of abstraction along the artefact development lifecycle (Buckl et al. 2010). Therefore, EA frameworks need to embrace change in ways that sufficiently consider the emerging new paradigms, such as the paradigm of service orientation (Alwadain et al. 2010; Postina et al. 2010) in order to deliver such benefits.

Service-Oriented Architecture (SOA) has gained much popularity in both academia and practice since it was introduced by a Gartner analyst in 1996. Service orientation is based on recognised concepts such as loose coupling, composition and coordination of building blocks and complexity reduction (Schroth 2007). Many definitions of SOA have been published by academics and practitioners. In general, the definitions can be classified into two perspectives: a technical and a business/managerial perspective. Although most of the definitions are predominantly technical, recent publications have taken a broader perspective of SOA from a business viewpoint (Lee et al. 2010). For example, Engels et al. (2008) define SOA as a paradigm for structuring the business of an enterprise in the form of services which accordingly steers the IT architecture. In fact, SOA provides a framework to assist communication and interaction between services. These services are published by service providers with related service level agreements in service registries to be accessed and utilised by service consumers (Luthria and Rabhi 2009). Thus, there are three key players in SOA: the service provider, the service consumers and the agencies that help consumers find services (Erl 2005; Luthria and Rabhi 2009; Papazoglou 2003).

The SOA Reference Model which was introduced by the public standardisation body OASIS (MacKenzie et al. 2006) describes SOA core elements such as service, contract, policy, service function, service provider, service consumer, service description, service interface, and the relationships between these elements. This model is used to understand the significant elements in a service-oriented environment and for the development of consistent standards and specifications that support the environment. Further, The Everware-CBDI Inc has introduced the CBDI Service Architecture and Engineering™ (CBDI-SAE™), which is an SOA meta-model that includes a taxonomy of all terms used (Everware-CBDI Inc 2009). It provides elements for describing a business-oriented SOA, independent of technology and services implementation. However, the CBDI-SAE™ has not been integrated with any existing enterprise architecture framework (Stein et al. 2008). Nonetheless, the meta-model has a first-class service element that executives relate to very well and contains some elements logically associated with it such as service interface, service dependency, service domain, participant and service classification. Service specification, which is used in the software realm, includes the elements that describe a service such as service definition, service operation, SLA, versioning, service state, and policy.

## RESEARCH DESIGN

This research consisted of four steps. First, the EA frameworks that would be within the scope of this study were identified. The second step was to review each framework with respect to their integration of SOA elements. In the third step, the identified elements were grouped in generic categories based on similarities. In the fourth step, the elements were compared across the different frameworks.

The first step was to select EA frameworks for this study. The adopted criteria to choose the EA frameworks were: (1) wide use and popularity, and (2) support of SOA elements. In terms of wide use, a recent survey from Infosys (2009) showed that TOGAF, Zachman and FEAF are the most adopted EA frameworks. Another survey showed that TOGAF, Zachman, FEAF, DoDAF, Gartner, and MODAF are the most used frameworks (Varnus

and Panaich 2009). In the literature, Leist and Zellner (2006) claimed that ARIS, DoDAF, FAEAF, MDA, TEAF, TOGAF and Zachman are reputable and widely deployed EAs. In addition, Winter and Fischer (2007) argued that TOGAF, FEAF and ARIS are widely used EA frameworks and compared them in terms of the essential elements in each framework.

The second criterion for EA frameworks selection in this study was the integration of SOA elements in these frameworks. The five frameworks incorporated in this study have documentations and/or provided meta-models describing how SOA elements are integrated in their EAs (see e.g., (Berrisford and Lankhorst 2009; Federal CIO Council 2008; US Department of Defence 2009). Therefore, TOGAF, FEAF, DoDAF and MoDAF were included in this study, because they meet these criteria. ArchiMate was also included, because it was developed as a service-oriented EA modelling language (Berrisford and Lankhorst 2009). The Zachman framework was excluded, because it was previously covered in another study (Alwadain et al. 2010). The Gartner framework was also excluded as it was not fully accessible due to commercial restrictions (Franke et al. 2009). Stein et al. (2008) showed how SOA modelling is enabled in ARIS. However, it did not explain how SOA elements fit into the original structure of the framework. Therefore, ARIS was excluded, too.

In the second step, the SOA elements in the selected frameworks were identified and presented in Table 1. When SOA elements were found explicitly in the frameworks' meta-models, then that was considered as very strong evidence for the explicit integration of SOA elements and “++” is used in the table to designate this. Similarly, if they were identified explicitly in the textual documentation, then this was considered strong evidence and “+” was used in the table. However, if they were found implicitly in the documentation (i.e., it was uncertain whether they are part of SOA or not), then this was seen as weak evidence of the integration and “+/-” was used in the table. The identification for each framework is described explicitly to make this process transparent and verifiable.

When we analysed the different approaches to integrating SOA elements across various EA frameworks, we noticed that there were conceptual as well as terminological variations with regard to the SOA elements that were represented in these frameworks. In order to be able to provide a more general overview and comparison of the approaches without losing detail, we decided to keep the original terms as used in the respective frameworks, but group them together in more generic categories based on the key elements of the OASIS reference model mentioned earlier. The categories are services, actors (e.g. service provider and consumer), service interfaces, service contract, and others for elements that could not be grouped together (e.g. service description, service policy, and service function). A discussion of the findings of the comparison is presented in section six. Both the original names of SOA elements and the grouping are presented. The columns of the table represent the five frameworks and only their layers (viewpoints) that have SOA elements, as some frameworks have represented SOA elements in one layer, e.g. DoDAF.

## SOA INTEGRATION IN EA FRAMEWORKS

This section outlines the examination and comparison of how SOA elements are integrated in the selected EA frameworks in terms of the completeness of their integration and their relative position in the EA layers (viewpoints) to cover stakeholders' concerns.

### ArchiMate

ArchiMate is an Enterprise Architecture modelling language. It has been introduced with a focus on service orientation and it defines the following three layers (viewpoints): the Business, Application and Technology. It models the global structure in each layer, the main artefacts, components, and dependencies between them, and the relationships between the layers (Iacob et al. 2007). Further, it has been adopted by the Open Group for modelling enterprise architecture (The Open Group 2009a). In order to identify SOA elements in ArchiMate, ArchiMate Specification (The Open Group 2009a) was investigated. First, in the business layer, a product which is defined as a coherent collection of services, a business service, a business interface and a contract are identified in the provided meta-model. However, the contract is associated with the product (collection of services), not the business service. SLA and QoS are recognised in the text as part of the contract. Second, in the application layer, an application service and an application interface are found in the meta-model. Third, in the technology layer, an infrastructure service and an infrastructure interface are recognised in the meta-model.

### The Open Group Architecture Framework (TOGAF)

The Open Group Architecture Framework (TOGAF) is considered an industry Enterprise Architecture framework. The recent rework of TOGAF for version 9 consists of six main parts. *The Architecture Development Method* (ADM) illustrates the ten different phases of EA development. *The ADM techniques and guidelines* are provided to support the application of the ADM and to deal with different scenarios and different process styles. *The content framework* provides a conceptual meta-model for describing architectural artefacts. It

is not considered compulsory and could be combined with other meta-models. *The enterprise continuum* is a virtual repository to maintain the architectural assets like models and architectural descriptions. *The TOGAF reference models* are divided into The TOGAF Technical Reference Model and the Integrated Information Infrastructure Reference Model. *The Architecture Capability Framework* is a set of the required skills, roles and responsibilities to establish and operate an Enterprise Architecture (Buckl et al. 2009; The open Group 2009c). SOA elements in TOGAF are represented in its meta-model and explained further in its documentation (the Open Group 2009b; The open Group 2009c). First, in the business architecture, a business service, a contract, service quality and a measure, which links objectives to business services, are identified in the meta-model. A service level agreement (SLA) is also found in the documentation. In the application architecture, an information system service is represented, and in the technology architecture, a platform service is identified in the meta-model. In TOGAF documentation, a service interface, service attributes and a service policy are identified as part of all three architectures of business, application and technology.

### **The Federal Enterprise Architecture Framework (FEAF)**

The Federal Enterprise Architecture Framework was developed by the US Federal Chief Information Officers Council. The FEAF was introduced to comply with the Clinger-Cohen Act to develop and maintain integrated systems architectures and to promote and organise Federal information sharing across the Federal government agencies (Urbaczewski and Mrdalj 2006). It has five reference models: the Performance Reference Model (PRM), the Business Reference Model (BRM), the Service Component Reference Model (SCRM), the Technical Reference Model (TRM) and the Data Reference Model (DRM). The reference models are designed to standardise terms and definitions in EA contexts and facilitate sharing and collaboration across the entire Federal government (Federal CIO Council 2008). In order to identify SOA elements in the FEAF, its SOA practical guide was investigated (Federal CIO Council 2008). However, no meta-model is provided. Thus, the documentation itself was used to evaluate the extent to which SOA elements and artefacts are integrated. The service model is apparent within the BRM and the SCRM. The following elements and artefacts of service portfolio, business service, IT service, Quality of Service, SLA, service contract, service consumer and service provider are recognised in the documentation.

### **The Department of Defence Architecture Framework (DoDAF)**

DoDAF V2.0 is a comprehensive, overarching framework and conceptual model that enables the development of architectures to assist DoD managers in making key decisions more effectively. The vision for DoDAF is to provide a comprehensive set of architecture concepts, methods and best practices to facilitate architecture development, to define and institutionalise the Net-Centric Data Strategy and Net-Centric Services Strategy of the Department, and to define, describe and develop services through the introduction of SOA. DoDAF V2.0 has different viewpoints: *Systems Viewpoint (SV)*, *Service Viewpoint (SvcV)*, *Data & Information Viewpoint (DIV)*, *Operational Viewpoint (OV)*, *Standards Viewpoint (StdV)*, *Capability Viewpoint (CV)*, *Project Viewpoint (PV)* and *All Viewpoints (AV)* (US Department of Defence 2009). The DoDAF documentation was reviewed to identify SOA elements. In the DoDAF generic meta-model, a service (including business and software services), service description, a service port and a service performer (both a service consumer and a service provider) are found. The DoDAF documentation contains service function, service contract (SLA is part of it), service policy, QoS, service channel. The main viewpoint that has SOA elements is the *Service Viewpoint (SvcV)*. However, these elements may appear in some other viewpoints such as *All Viewpoints* and *Capability Viewpoint* when mapping services to capabilities.

### **The Ministry of Defence Architecture Framework (MODAF)**

The MODAF is an enterprise architecture framework developed by the UK Ministry of Defence to support its planning and management activities. The MODAF provides a consistent set of rules and templates, known as views, which provide a textual and graphical visualisation of an area of the business. Each view provides a different perspective on the business to match different stakeholder interests. The views are divided into seven categories: strategic views, operational views, service oriented views, systems views, acquisition views, technical views and all views. The MODAF offers a meta-model which defines the relationship between all the data in all the views (The UK Department of Defence 2010a). The service-oriented viewpoint provides seven views to provide a perspective that enables service specification, behaviour and policies. The views do not focus on the detailed implementation of services, but on requirements that the services fulfil. A service, service interface, SLA, service policy, service function, service attributes (description), and service consumer have been identified in the MODAF's models as SOA related elements (The UK Department of Defence 2010b).

[illegible]

## COMPARISON OF SOA INTEGRATION IN EA FRAMEWORKS

The five selected EA frameworks were compared in order to provide an overview of the commonalities and differences in terms of the SOA elements covered and the position of these elements in the layers (viewpoints) of the five frameworks. Conclusions based upon this comparison are presented in the next section.

First is the *services category*. The service is found in all the frameworks; however, it differs substantially in the details. For example, a generic service element is identified in the meta-models of DoDAF V2.0 and the MODAF, while a business service is recognised in the meta-models of ArchiMate and TOGAF, and in the documentations of the FEAF and DoDAF V2.0. In addition, an application service is identified in the meta-models of ArchiMate and DoDAF V2.0. Further, an Information System service is recognised in TOGAF's meta-models, and an enterprise service is identified in the FEAF's documents. Further, an infrastructure service is found in ArchiMate's meta-model, while a platform service is identified in TOGAF's meta-model.

Second is the *actors' category*. In ArchiMate's and TOGAF's meta-models, an actor is represented in the business layer. A service provider is found in the FEAF's documents in BRM and SCRM, while a performer, which could be a service provider or a consumer, is identified in the MODAF's meta-models.

Third is the *service interfaces category*. In ArchiMate's meta-models, a business interface is found in the business layer, an application interface is found in the application layer, and an infrastructure interface is found in the technology layer. All these interfaces are linked to the services in the same layer. However, in the meta-model of DoDAF V2.0, it is called a service port, while in the documents of TOGAF and the FEAF, and in the MODAF's meta-models, it is called a service interface.

Fourth is the *service contracts category*. A contract is recognised in ArchiMate's meta-models in the business layer, in TOGAF's meta-models in the business architecture, in the FEAF's documents in BRM and SCRM, and in the DoDAF V2.0 documents. However, it was noticed that the contract is associated with the product, which is a collection of services, in ArchiMate. An SLA is identified in ArchiMate's meta-models in the business layer as part of the contract and in TOGAF's document as part of the contract, too. It is also recognised in the FEAF's document in BRM and SCRM viewpoints and in DoDAF's services viewpoint. The SLA is found in the meta-models of the MODAF's service-oriented viewpoint. The service conditions element is identified in ArchiMate's meta-models as a part of the contract. In contrast, it is found in the DoDAF's documents as part of the service description in the services view. However, it is not mentioned in other frameworks. The Quality of Service (QoS) is identified in ArchiMate's document in the business layer as part of the contract and TOGAF's meta-model in the business architecture. It is also recognised in the FEAF's document in both BRM and SCRM viewpoints and in the DoDAF's documents in the services view.

Finally, there are a couple of single elements that are not grouped. First, a product which is defined as a coherent collection of services accompanied by a contract is found only in ArchiMate's meta-model in the business layer. Next, a measure element, which links the objective and business service, is only identified in TOGAF's meta-models in business architecture. Then, a service description is found in TOGAF's document in all three architectures, in the DoDAF's and the MODAFs' meta-models. Next, a service policy is identified in TOGAF's document in all the three layers, in the DoDAF's documents and in the MODAF's meta-models. Then, a service channel, which is a logical or physical communication path between requisitions and services, is recognised only in the DoDAF's meta-models. Finally, a service function is only found in the DoDAF's documents and in MODAF's meta-models.

## DISCUSSION AND CONCLUSIONS

The above analysis has resulted in a number of interesting findings. First, all the frameworks have a service element; however, by delving deeper, some differences were noticed. A business service is covered in most of the frameworks. However, the IS service, the enterprise service, the infrastructure service, and the platform service are less frequent. Each of these is only covered in one framework, although some of them might mean the same thing, e.g., an infrastructure service and a platform service. Further, it is clear that TOGAF and ArchiMate have a clear classification and representation of services in all their layers (viewpoints).

Second, the actor element is recognised in the frameworks. A generic element called actor was found in ArchiMate and TOGAF to represent both the service provider and the consumer. The separation of the provider from the consumer in two elements could only be found in the FEAF. The actor element is similar to many of the other elements in terms of the inconsistency of terminology and whether a generic actor element could be used to represent both the provider and the consumer.

Third, in terms of the service interface, all the frameworks cover interfaces as part of SOA. However, interface-related elements are represented through different terms in the various frameworks. For example, in the DoDAF, the term "service port" was chosen instead of "service interface", and in ArchiMate different names are given to



interfaces depending on the layer of their associated service (e.g., a business interface of a business service). Fourth, the contract, SLAs and QoS are also well covered in all the frameworks except the MODAF, which only covers SLAs. In some cases, it is not clear whether an SLA is part of or equal to the contract.

The service description and service policy are missing in ArchiMate and FEAF, while the service function is only given in the DoDAF and the MODAF. The SOA element product is found only in ArchiMate to represent a group of services. A measure is found only in TOGAF and a service channel is only identified in the DoDAF. It seems that for the integration of SOA elements into the EA frameworks, the product, measure and service channel were considered less important elements of SOA by the intellectual creators of these integrations, as we have not identified them in more than one framework of the five, nor are they featured in the OASIS SOA reference model or the CBDI-SAE<sup>TM</sup>.

Further, it appears that the existing frameworks can generally accommodate the elements that constitute the paradigm of service-orientation. However, there is no uniformity with regard to the SOA elements and their level of details in the EA frameworks. In addition, although ArchiMate has a particularly strong focus on service-orientation and was adopted by the Open Group and the TOGAF body, it is still incomplete with regard to the SOA elements such as service descriptions and the representation of specific types of services such as enterprise services. SOA elements and concerns for stakeholders are still unclearly supported by the variations in these selected EA frameworks. This finding supports what Luo (2006) claimed that there are inconsistency in terms of EA terminologies and concepts which leads to confusion in communications within the EA discipline and with EA stakeholders. Luo (2006) added that such inconsistency will not advance the discipline. Given that EA suffers from lack of consistency in terminology and concepts, the gap is even getting bigger by incorporating SOA which lacks consistency as well. Consequently that might impact the use of EA as communication tool between different stakeholders. In addition, as discussed earlier in the introduction, the definition of SOA and its scope is still confusing. SOA has emerged as an IT architecture style and has evolved to cover both business and IT moving towards a Service-Oriented Enterprise which might have an impact on the integration of SOA in EAs. Furthermore, the lack of consistency in terms of SOA terminology, scope and abstraction layers (Viering et al. 2009) is evident in the way EAs integrated SOA. The findings of this paper are supported by our earlier findings from the comparison of different types of SOA integration in the Zachman framework (Alwadain et al. 2010). It was concluded that even on a single EA framework level; there are variations in terms of SOA integration.

In this paper, five EA frameworks were investigated and compared. They all support service-orientation at different levels. There is inconsistency in what elements of SOA should be captured and where they should be represented (viewpoints) in order to fulfil the stakeholders' SOA-related concerns. Given that EA elements and artefacts have to be relevant for a wide range of enterprises and government agencies, SOA integration in existing EA frameworks needs to be addressed to precisely identify what elements are required for such integration in order to (1) reduce the ambiguity surrounding the level of integration (viewpoints of EA) and the depth of the integration (actual SOA elements) to address SOA stakeholders' concerns and (2) to help practitioners design service-oriented "as-is" or "to-be" models.

The contribution of the paper is that it is the first paper to compare EA frameworks in regard to SOA elements. Further, this study is merely the first step in understanding and improving the integration of SOA in EA. While the comparison does not resolve any ambiguity, it does point to the issues and provide examples of how the different frameworks tackle the integration. Future research needs to look beyond existing literature to better identify the role of service and SOA for EA and thereby better assist in the determination of SOA elements in an EA. This work is in response to Viering et al.'s (2009) call for more research on how SOA changes EA frameworks and Greefhorst et al.'s (2006) call for more work on architecture description standardization. In future research, this work will be extended using a qualitative study to identify SOA elements that should be captured in EA and to explain the differences in the integration approaches which could be generalized beyond services and service orientation to new emerging concepts and paradigms that need to be integrated in EA frameworks. However, this study has limitations too. For practical reasons, it only focused on the SOA elements and not on their relationships and interactions with other elements in EA frameworks. If all the interactions and relationships had been included, the comparisons would have become complex and hard to maintain in a paper like this, given the space restriction. The study also ignored some elements in some EA frameworks that might be considered similar to certain SOA elements unless explicit statements in these frameworks explained them, e.g., some practitioners may equalize a function to a service. In addition, our comparison of EA frameworks is explorative so far and lacks a solid theoretical foundation. This is a common limitation of EA research (Schmidt and Buxmann 2011; Tamm et al. 2011).

## REFERENCES

- Alwadain, A., Korthaus, A., Fiel, E., and Rosemann, M. 2010. "Integrating Soa into an Enterprise Architecture – a Comparative Analysis of Alternative Approaches," in: *The Fifth IFIP International Conference on Research and Practical Issues of Enterprise Information Systems (CONFENIS)*. Natal, Brazil.
- Berrisford, G., and Lankhorst, M. 2009. "Using Archimate with Togaf." Retrieved October 20, 2010, from <http://www.via-nova-architectura.org/artikelen/tijdschrift/using-archimate-with-togaf.html>
- Buckl, S., Ernst, A.M., Matthes, F., Ramacher, R., and Schweda, C.M. 2009. "Using Enterprise Architecture Management Patterns to Complement Togaf.," *IEEE International Enterprise Distributed Object Computing Conference*, pp. 34-41.
- Buckl, S., Matthes, F., Schulz, C., and Schweda, C.M. 2010. "Exemplifying a Framework for Interrelating Enterprise Architecture Concerns," in *Ontology, Conceptualization and Epistemology for Information Systems, Software Engineering and Service Science*, M.-A. Sicilia, C. Kop and F. Sartori (eds.). Springer Berlin Heidelberg, pp. 33-46.
- Chen, D., Doumeingts, G., and Vernadat, F. 2008. "Architectures for Enterprise Integration and Interoperability: Past, Present and Future," *Computers in Industry* (59:7), pp. 647-659.
- Correia, A., and Silva, M. 2007. "Modeling Services in Information Systems Architectures.," in *Digital Enterprise Technology Perspectives and Future Challenges* P.F. Cunha and P.G. Maropoulos (eds.). New York, USA: Springer., pp. 157-164.
- Engels, G., Hess, A., Humm, B., Juwig, O., Lohmann, M., Richter, J.P., Voß, M., and Willkomm, J. 2008. "A Method for Engineering a True Service-Oriented Architecture.," *Proceedings of the 10th International Conference on Enterprise Information Systems (ICEIS 2008)*. Barcelona, Spain., pp. 272-281.
- Erl, T. 2005. *Service-Oriented Architecture : Concepts, Technology, and Design*. New York: Prentice Hall PTR.
- Everware-CBDi Inc. 2009. "Cbdi-Sae Meta Model for Soa V3," in: *CBDi Journal*.
- Federal CIO Council. 2008. "A Practical Guide to Federal Service Oriented Architecture Version 1.1."
- Franke, U., Hook, D., König, J., Lagerstrom, R., Narman, P., Ullberg, J., Gustafsson, P., and Ekstedt, M. 2009. "Eaf2- a Framework for Categorizing Enterprise Architecture Frameworks.," *Proceedings of the 10th ACIS International Conference on Software Engineering, Artificial Intelligences, Networking and Parallel/Distributed Computing.*: IEEE Computer Society, pp. 327-332.
- Greefhorst, D., Koning, H., and Vliet, H. 2006. "The Many Faces of Architectural Descriptions," *Information Systems Frontiers* (8:2), pp. 103-113.
- Grigoriu, A. 2007. "Soa, Bpm, Ea, and Service Oriented Enterprise Architecture." Retrieved 13 Aug, 2010, from <http://www.bptrends.com/publicationfiles/12-07-ART-Service%20Oriented%20Enterprise%20Architecture-Grigoriu-final-Use.pdf>
- Gustas, R. 2007. "Semantic Integration in Conceptual Modeling of Service Architectures," *Transactions on Internet Research* (3:2), pp. 3-7.
- Jacob, M.E., Jonkers, H., Lankhorst, M.M., and Steen, M.W.A. 2007. "Service-Oriented Enterprise Modelling and Analysis: A Case Study," *International Journal of Business Process Integration and Management* (2:1), pp. 26-37.
- Infosys. 2009. "Enterprise Architecture Expands Its Role in Strategic Business Transformation, Infosys Enterprise Architecture Survey 2008/2009." Retrieved February, 05, 2010, from <http://www.infosys.com/offering/IT-services/architecture-services/ea-survey/Pages/index.aspx>
- Khoshnevis, S., Aliee, F.S., and Jamshidi, P. 2009. "Model Driven Approach to Service Oriented Enterprise Architecture.," *The 2009 IEEE Services Computing Conference.*, pp. 279-286.
- Knippel, R., and Skytte, B. 2007. "Service Oriented Enterprise Architecture: An Opportunity of Evolution," *Journal of Enterprise Architecture* (3:3), pp. 17-25.
- Kumar, S., Dakshinamoorthy, V., and Krishnan, M.S. 2007. "Does Soa Improve the Supply Chain? An Empirical Analysis of the Impact of Soa Adoption on Electronic Supply Chain Performance.," *Proceedings of the 40th Hawaii International Conference on System Sciences*.
- Lankhorst, M. 2005. *Enterprise Architecture at Work: Modelling, Communication and Analysis*. Springer-Verlag GmbH.
- Lee, J.H., Shim, H.-J., and Kim, K.K. 2010. "Critical Success Factors in Soa Implementation: An Exploratory Study," *Information Systems Management* (27:2), pp. 123 - 145.
- Leist, S., and Zellner, G. 2006. "Evaluation of Current Architecture Frameworks.," *The 2006 ACM symposium on Applied computing*, pp. 1546-1553.
- Luo, H. 2006. "Building Ea Management Standards:The Progresses and Challenges of the Ea Management Guide Project." a|EA Standards Committee.
- Luthria, H., and Rabhi, F. 2009. "Service Oriented Computing in Practice - an Agenda for Research into the Factors Influencing the Organizational Adoption of Service Oriented Architectures," *Journal of Theoretical and Applied Electronic Commerce Research* (4:1), pp. 39-56.
- MacKenzie, C.M., Laskey, K., McCabe, F., Brown, P.F., and Metz, R. 2006. "Reference Model for Service Oriented Architecture 1.0. Technical Report."
- Mykhashchuk, M., Buckl, S., Dierl, T., and Schweda, C.M. 2011. "Charting the Landscape of Enterprise Architecture Management.," in: *10th International Conference on Wirtschaftsinformatik.* Zurich, Switzerland.
- Papazoglou, M.P. 2003. "Service-Oriented Computing: Concepts, Characteristics and Directions," in: *Fourth International Conference on Web Information Systems Engineering*. Rome, Italy.

- Paras, G., DeBoever, L., and Westbrook, T. 2007. "Ea Practitioners 'Cheatsheet': 12 Ideas You Can Use Today to Improve Your Ea Team's Effectiveness." Retrieved 13 Aug, 2010, from <http://www.eadirections.com/uploads/file/EAdirections%2012%20Ideas%20Webinar%20Nov-13-07%20Final.pdf>
- Perko, J. 2008. "It Governance and Enterprise Architecture as Prerequisites for Assimilation of Service-Oriented Architecture. Phd Thesis.." Tampere University of Technology.
- Postina, M., Trefke, J., and Steffens, U. 2010. "An Ea-Approach to Develop Soa Viewpoints. ," *14th IEEE International Enterprise Distributed Object Computing Conference.*, pp. 37-46.
- Sanders, D.T., Hamilton, J.A., and MacDonald, R.A. 2008. "Supporting a Service-Oriented Architecture," in: *Proceedings of the 2008 Spring simulation multiconference.* Ottawa, Canada: Society for Computer Simulation International, pp. 325-334
- Schmidt, C., and Buxmann, P. 2011. "Outcomes and Success Factors of Enterprise It Architecture Management: Empirical Insight from the International Financial Services Industry," *European Journal of Information Systems* (20:2), pp. 168-185.
- Schroth, C. 2007. "The Service-Oriented Enterprise," *Journal of Enterprise Architecture* (3:4), pp. 73-80.
- Seppänen, V. 2008. "Interconnections and Differences between Ea and Soa in Government Ict Development.," *Proceedings of the 31st Information Systems Research Seminar in Scandinavia.*, Åre, Sweden.
- Sowa, J.F., and Zachman, J.A. 1992. "Extending and Formalizing the Framework for Information Systems Architecture," *IBM Systems Journal* (31:3), pp. 590-616.
- Sprott, D. 2011. "Bridging the Gap between Abstract Soa Models and Reality." Retrieved March, 2nd, 2011, from <http://davidsprottsblog.blogspot.com/2011/02/bridging-gap-between-abstract-soa.html>
- Stein, S., Lauer, J., and Ivanov, K. 2008. "Aris Method Extension for Business-Driven Soa," *WIRTSCHAFTSINFORMATIK* (50:6), pp. 436-444.
- Tamm, T., Seddon, P.B., Shanks, G., and Reynolds, P. 2011. "How Does Enterprise Architecture Add Value to Organisations?," *Communications of the Association for Information Systems* (28:1), p. 10.
- The Open Group. 2009a. *Archimate 1.0 Specification*. Zaltbommel, Netherlands.: Van Haren Publishing.
- the Open Group. 2009b. "Architectural Artifacts." Retrieved 15 October, 2010, from <http://www.opengroup.org/architecture/togaf9-doc/arch/index.html>
- The open Group. 2009c. "Content Metamodel " Retrieved 15 October, 2010, from <http://www.opengroup.org/architecture/togaf9-doc/arch/chap34.html>
- The UK Department of Defence. 2010a. "Mod Architecture Framework (Modaf)." Retrieved 22 October, 2010, from <http://www.mod.uk/DefenceInternet/AboutDefence/WhatWeDo/InformationManagement/MODAF/ModafDetailedGuidance.htm>
- The UK Department of Defence. 2010b. "The Modaf Service Oriented Viewpoint." Retrieved 24 October, 2010, from <http://www.mod.uk/DefenceInternet/AboutDefence/CorporatePublications/InformationManagement/MODAF/TheServiceOrientedViewpointsov.htm>
- Urbaczewski, L., and Mrdalj, S. 2006. "A Comparison of Enterprise Architecture Frameworks," *Issues in Information Systems* (7:2), pp. 18-23.
- US Department of Defence. 2009. "Dod Architecture Framework Version 2.0." Retrieved 20 October, 2010, from <http://cio-nii.defense.gov/sites/dodaf20/>
- Varnus, J., and Panaich, N. 2009. "Togaf 9 Survey Results." Retrieved January, 29, 2010, from <https://www.opengroup.org/conference-live/uploads/40/20081/Mon - am - 3 - Varnus.pdf>
- Viering, G., Legner, C., and Ahlemann, F. 2009. "The (Lacking) Business Perspective on Soa-Critical Themes in Soa Research. ," *Proceedings of Wirtschaftsinformatik.* , pp. 45-54.
- Weerakkody, V., Janssen, M., and Hjort-Madsen, K. 2007. "Integration and Enterprise Architecture Challenges in E-Government: A European Perspective," *International Journal of Cases on Electronic Commerce* (3:2), pp. 13-35.
- Winter, R., and Fischer, R. 2007. "Essential Layers, Artifacts, and Dependencies of Enterprise Architecture," *Journal of Enterprise Architecture* (3:2), pp. 7-18.
- Zachman, J.A. 1987. "A Framework for Information Systems Architecture," *IBM Systems Journal* (26:3), pp. 454-470.

## COPYRIGHT

Alwadain, Fielt, Korthaus and Rosemann © 2010. The authors assign to ACIS and educational and non-profit institutions a non-exclusive licence to use this document for personal use and in courses of instruction provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive licence to ACIS to publish this document in full in the Conference Papers and Proceedings. Those documents may be published on the World Wide Web, CD-ROM, in printed form, and on mirror sites on the World Wide Web. Any other usage is prohibited without the express permission of the authors.